SURROGATE GRADIENTS MISALLOCATE ADVERSARIAL SENSITIVITY IN SPIKING NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Spiking neural networks (SNNs) offer a promising basis for low-power intelligence through binary events. This discrete nature makes backpropagation-based training reliant on surrogate gradients. However, conventional surrogate functions with fixed profiles fail to adapt to decision-relevant neuronal dynamics under adversarial perturbations, limiting the effectiveness of robust learning in SNNs. Through systematic comparative experiments, we reveal how this lack of adaptation manifests as adversarial sensitivity misallocation, directing backward sensitivity away from membrane states involved in perturbation-induced spike flips. We therefore propose Flip-Aware Reallocation (FARE), which derives dynamic response ranges from decision-weighted spike flips. By making surrogate gradients aware of perturbation-induced flips, FARE reallocates backward sensitivity to bring the learning signal into closer alignment with the network's decision-relevant adversarial response. Across static, sequential, event-based, and large-scale tasks, FARE outperforms existing state-of-the-art methods in both adversarial robustness and clean accuracy. The code is provided in the supplementary materials.

1 INTRODUCTION

Spiking neural networks (SNNs) communicate through binary spikes generated by the temporal evolution of membrane potentials, offering a promising computational foundation for low-power intelligence (Roy et al., 2019; Davies et al., 2018; Esser et al., 2015). Advances in surrogate gradient learning have enabled high-accuracy training of SNNs through backpropagation despite their discrete computations (Neftci et al., 2019), supporting applications in classification (Fang et al., 2021b;a), sequence processing (Fang et al., 2023; Yin et al., 2021), and edge computing (Cramer et al., 2022).

With growing security demands in practical applications, maintaining stable decisions under adversarial perturbations has become an urgent necessity for trustworthy SNNs (Bu et al., 2023; Kundu et al., 2021). To improve this stability, adversarial training (Kundu et al., 2021; Ding et al., 2022), robustness-oriented learning (Ding et al., 2022; Liu et al., 2024), and neural modification (Ding et al., 2024a; Xu et al., 2024; Ding et al., 2024b) have been explored. Despite their different starting points, these approaches share a fundamental optimization requirement: backpropagation must guide targeted parameter updates through shifting part of the learning emphasis from fitting unperturbed samples toward improving resilience to adversarial perturbations (Zhang et al., 2019). The sensitivity signals underlying these updates are shaped by the surrogate functions in surrogate gradient learning (Neftci et al., 2019; Bu et al., 2023). Common choices, such as Triangle, Sigmoid, and Rectangle functions, directly allocate local sensitivity to membrane states using predefined fixed profiles (Wu et al., 2018).

This fixed design allocates backward sensitivity through simple, uniform local rules, yet struggles to reflect neuronal dynamics under adversarial perturbations. In spiking neurons, a small perturbation-induced membrane displacement near the firing threshold can trigger a spike flip, affecting the final decision through downstream transmission and subsequent state evolution (Neftci et al., 2019; Ding et al., 2024a). This transition from small quantitative shifts to qualitative changes means that adversarial sensitivity depends not only on threshold distance, but also on whether a perturbation induces a flip and how that flip affects the decision (Wang et al., 2026b). However, fixed surrogate profiles do not adapt their sensitivity allocation to this information (Wu et al., 2018). This disconnect can shift the states emphasized by backpropagation away from those most relevant to adversarial damage, limiting the targeted reinforcement of vulnerable network components during robust learning.

To systematically uncover this disconnect, we examine the model-level adversarial response and then trace the discrepancy to its internal sensitivity allocation. For the same adversarial perturbation, we

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

empirically observe widespread discrepancies between the decision damage estimated by surrogate gradients and that realized by the forward pass. Even with an optimal scaling factor fitted separately for each configuration, global rescaling explains only a small part of the discrepancy and cannot recover sample-wise variations in realized damage. After removing the influence of overall response scale, we further observe that surrogate responses and actual spike flips remain concentrated in different membrane-state regions. This consistent mismatch identifies sensitivity allocation, rather than magnitude, as the key source of the decision damage discrepancies, revealing our central finding:

Surrogate gradients misallocate adversarial sensitivity in spiking neural networks.

This misallocation arises precisely because fixed surrogate functions allocate backward sensitivity without feedback from perturbation-induced spike flips and their influence on the decision. To bridge this disconnect, we propose Flip-Aware Reallocation (FARE), which reallocates backward sensitivity based on dynamic feedback from actual spike flips. During training, FARE collects flip locations and weights them by their decision relevance to dynamically determine adaptive response membrane ranges. By combining this dynamic response with the original profile, FARE reallocates the sensitivity mass toward perturbation-reachable, decision-relevant flip regions while preserving threshold-local learning signals, enabling more targeted reinforcement of vulnerable network components with SNNs.

We evaluate FARE across diverse experimental configurations. FARE substantially improves white-box and black-box robustness while minimizing the sacrifice in clean accuracy. Mechanistic analyses and ablations further support the benefits of sensitivity reallocation and the complementary roles of the dynamic and original responses. Overall, the contributions of this paper can be summarized as:

- **We uncover a critical limitation of surrogate gradients.** We empirically identify adversarial sensitivity misallocation in surrogate gradients, revealing a limitation that hinders robust learning and highlighting sensitivity allocation as a key design principle for robust SNNs.
- **We introduce sensitivity reallocation driven by actual spike behavior.** FARE translates decision-relevant spike flips into dynamical adaptive response ranges, enabling surrogate functions to reallocate their sensitivity distribution while preserving total sensitivity mass.
- **We demonstrate SOTA robustness with limited clean-accuracy trade-offs.** FARE outperforms SOTA methods on multiple adversarial settings while largely retaining clean accuracy, with consistent robustness gains across diverse tasks and mainstream surrogate functions.

2 PRELIMINARY

Spike dynamics. The most widely used neuron model in SNNs is the Leaky Integrate-and-Fire (LIF) (Fang et al., 2021a). From an implementation perspective, the dynamics of a single LIF neuron are given in Eq. (1), where $u[t]$, $s[t]$, and $c[t]$ denote the membrane potential, the binary spike output, and the input current at time step t, respectively. Here, τ is the membrane time constant, u_{th} is the firing threshold, and H is the Heaviside function.

$$\begin{cases} u[t+1] = (1-\frac{1}{\tau})\big(u[t] - u_{\text{th}} s[t]\big) + c[t], \\ s[t+1] = H\big(u[t+1] - u_{\text{th}}\big), \end{cases} \tag{1}$$

$$\delta^\star \in \arg\max_{\|\delta\|_p \le \varepsilon} \mathcal{L}\big(f(x+\delta), y\big). \tag{2}$$

Adversarial attacks. Given an input x with label y, an adversarial perturbation $\delta^\star$ is obtained by maximizing the classification loss within an ℓ_p-norm ball of radius ε, as formulated in Eq. (2). The corresponding adversarial example is $x_{\text{adv}} = x + \delta^\star$. Detailed attacks are described in Appendix B.

3 MOTIVATION

Adversarial robustness is ultimately evaluated by prediction stability under bounded perturbations, while the decision margin provides a more informative measure of how close a model is to adversarial failure (Goodfellow et al., 2014; Madry et al., 2017; Zhang et al., 2019). In gradient-based adversarial learning, the input gradient provides a first-order description of how perturbations around a sample change the loss and decision margin (Kurakin et al., 2016; Carlini & Wagner, 2017). Gradients therefore serve not only as optimization tools, but also as sensitivity signals for local adversarial vulnerability (Ross & Doshi-Velez, 2018; Madry et al., 2017). For differentiable ANNs, this sensitivity signal follows directly from the model's local derivative. In SNNs, however, the forward

computation follows the hard-spike rule as Eq. (1) shows, whereas the backward propagation replaces its non-differentiable derivative with a surrogate function $\psi(\cdot)$ (Neftci et al., 2019; Wu et al., 2018). Consequently, the sensitivity signal available to adversarial attacks and training in SNNs is not the true derivative of the hard-spike dynamics, but an estimate induced by the chosen surrogate function.

This forward-backward separation naturally manifests as a numerical discrepancy between the local damage estimated by the surrogate gradient and the finite-step adversarial damage actually realized by the hard-spike dynamics. Although exact agreement is neither expected nor required for any nonlinear models (Madry et al., 2017), this discrepancy takes on a distinct dynamical meaning in SNNs. An SNN is naturally a hybrid dynamical system that couples continuous membrane evolution with discrete spike emission and threshold-triggered reset. A small membrane displacement near the firing threshold can flip the binary spike state, injecting or suppressing an event that changes downstream synaptic input and subsequent state trajectories. These microscopic local changes can accumulate across layers and timesteps into a substantial macroscopic shift in the decision margin. Therefore, the forward-backward damage residual provides a model-level diagnostic of whether the local sensitivity view supplied by the surrogate gradient adequately characterizes the finite-step response of the hard-spike network. To quantify this residual, we compare two views of the damage caused by the same adversarial perturbation: the finite-step margin damage realized by the hard-spike forward pass and its first-order estimate obtained from the clean-point surrogate gradient, as follows.

Definitions of adversarial damage and residual. Let $f(x) \in \mathbb{R}^C$ denote the logits produced by the hard-spike forward pass after temporal readout, where C is the number of classes. For an input x with label y, we define the classification margin as $M(x, y) = f_y(x) - \max_{j \neq y} f_j(x)$. A larger margin indicates greater logit separation between the correct class and its strongest competitor. Given a fixed adversarial perturbation $\delta^\star$ from Eq. (2), the realized finite-step adversarial damage is obtained by

$$D_{\text{real}}(x, \delta^\star) = M(x, y) - M(x + \delta^\star, y). \tag{3}$$

At the clean input, the surrogate backward rule provides the input sensitivity $\widetilde{\nabla}_x^\psi M(x, y)$, namely the gradient of the classification margin with respect to the input computed using ψ during backward propagation. Its first-order damage estimate along the same perturbation follows Eq. (4), where $\langle \cdot, \cdot \rangle$ denotes the inner product. The negative sign converts the margin change into margin damage.

$$D_{\text{est}}^\psi(x, \delta^\star) = -\left\langle \widetilde{\nabla}_x^\psi M(x, y), \delta^\star \right\rangle. \tag{4}$$

Additionally, for a fixed $\delta^\star$, the magnitude of D_{est}^ψ changes under multiplicative rescaling of the surrogate derivative, whereas D_{real} is determined solely by the hard-spike forward pass. We therefore compare surrogate functions under the unit-mass convention $\int_{-\infty}^{+\infty} \psi(z)\,\mathrm{d}z = 1$, where $z = u - u_{\text{th}}$ is the membrane-to-threshold distance. This convention follows from viewing ψ as the derivative of a unit-height smooth approximation to the Heaviside function, thereby removing an arbitrary neuron-level scale factor. Under this convention, we define the forward-backward damage residual as

$$R_\psi(x, \delta^\star) = D_{\text{real}}(x, \delta^\star) - D_{\text{est}}^\psi(x, \delta^\star). \tag{5}$$

Residual examination. We examine whether the damage residual is an isolated artifact of a few configurations or a recurring model-level phenomenon across SNNs. We construct 108 experimental cells on static (CIFAR-10), sequential (SeqCIFAR-10), and event-based (DVSCIFAR-10) datasets by combining three surrogate functions, three timestep settings, and four perturbation budgets. Full experimental details are provided in Appendix E. Valid cells in Figure 1 reveal pronounced forward-backward discrepancies across three datasets. Both the magnitude and sign of the residual vary substantially with timestep setting, surrogate function, and perturbation budget. In particular, different surrogate functions produce sharply different residual patterns even when all other settings are fixed. These results establish a widespread and strongly configuration-dependent separation between the local sensitivity view induced by the surrogate gradient and the response of the hard-spike network. This observation raises a more intuitive structural question: Can this discrepancy be simply explained, or, eliminated by a global scaling of the surrogate gradient?

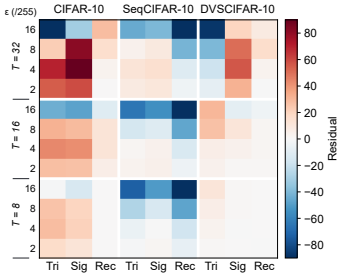

Figure 1: Damage residual cells.

Global scaling does not explain the discrepancy. To give the scaling hypothesis its most favorable test, we split the correctly classified samples from each configuration into calibration and held-out sets. On the calibration set, we independently fit an optimal positive scalar as Eq. (6). We then evaluate the relative reconstruction error on the held-out set as Eq. (7), where $\mathrm{Err}(c) = 0$ indicates exact reconstruction of the realized damage.

$$c^\star = \arg\min_{c>0} \sum_i \left(D_{\mathrm{real},i} - c D^{\psi}_{\mathrm{est},i} \right)^2 . \quad (6)$$

$$\mathrm{Err}(c) = \frac{\left\| D_{\mathrm{real}} - c D^{\psi}_{\mathrm{est}} \right\|_2}{\| D_{\mathrm{real}} \|_2} . \quad (7)$$

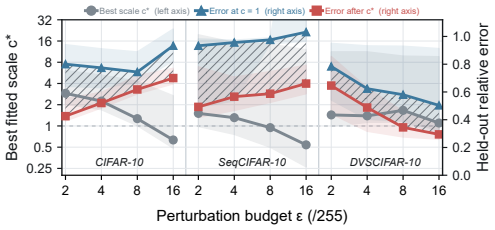

Figure 2: Global scales and held-out errors.

Figure 2 reports the best fitted scale $c^\star$, the error without rescaling $\mathrm{Err}(1)$, and the error after optimal rescaling $\mathrm{Err}(c^\star)$. If the discrepancy primarily arose from the overall magnitude of the surrogate gradient, $c^\star$ would remain relatively stable across configurations and reduce the held-out error toward zero. Instead, its median varies from 0.54 to 2.90 across datasets and perturbation budgets, with broad interquartile ranges. More importantly, even after fitting a separate optimal scale for every configuration and budget, the median held-out error remains between 0.29 and 0.70. In short, even under the most favorable setting, global rescaling corrects only part of the magnitude error and still fails to recover the sample-wise structure of finite-step damage. Because a scalar uniformly changes the magnitude of the surrogate response without altering its relative sensitivity pattern, the remaining discrepancy cannot be explained by numerical scale alone, instead, it points to a structural mismatch in how surrogate sensitivity is allocated across membrane states relative to the hard-spike response.

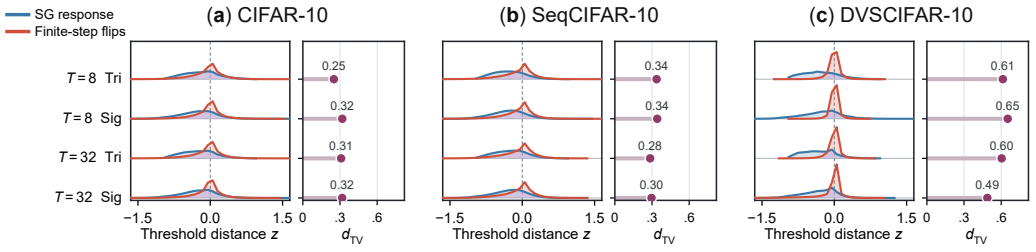

Figure 3: Scale-free sensitivity allocation across membrane states.

Scale-free comparison of surrogate response. Having ruled out global scaling as a sufficient explanation, we remove magnitude from the comparison and examine where the surrogate response is allocated across membrane states. Let k index a neuron at a particular sample, we denote its clean membrane-to-threshold distance, adversarial membrane displacement, and finite-step spike flip by

$$z_k = u_k - u_{\mathrm{th}}, \qquad q_k = z_k(x + \delta^\star) - z_k(x), \qquad \Delta s_k = H(z_k + q_k) - H(z_k). \quad (8)$$

For the same displacement q_k, the surrogate backward rule assigns the local response mass $a^{\psi}_k = |\psi(z_k) q_k|$, whereas the hard-spike forward pass realizes the flip mass $b_k = |\Delta s_k|$. We aggregate these quantities over bins B of the clean threshold distance and normalize them separately:

$$P_{\mathrm{SG}}(B) = \frac{\sum_{k:z_k \in B} a^{\psi}_k}{\sum_k a^{\psi}_k}, \qquad P_{\mathrm{flip}}(B) = \frac{\sum_{k:z_k \in B} b_k}{\sum_k b_k}. \quad (9)$$

Here, P_{SG} describes where the surrogate rule places its response to the observed adversarial displacement, while P_{flip} records where the hard-spike dynamics actually change state. Since both profiles have unit mass, multiplying the surrogate gradient by any positive constant leaves P_{SG} unchanged. Their allocation discrepancy can therefore be measured independently of global scale using total variation distance d_{TV} as Eq. (10). A value of $d_{\mathrm{TV}} = 0$ indicates identical allocation, whereas a larger value represents less shared mass between the two profiles.

$$d_{\mathrm{TV}} = \frac{1}{2} \sum_B |P_{\mathrm{SG}}(B) - P_{\mathrm{flip}}(B)| = 1 - \sum_B \min\{P_{\mathrm{SG}}(B), P_{\mathrm{flip}}(B)\} . \quad (10)$$

Figure 3 presents VGG11 results under FGSM with $\varepsilon = 8/255$. Across all settings, d_{TV} ranges from 0.25 to 0.65, showing that the misallocation remains substantial after scale has been eliminated. More importantly, within each dataset, the response profiles remain broadly similar and d_{TV} stays consistently nonzero when the timestep or surrogate function is changed: 0.25-0.32 on CIFAR-10, 0.28-0.34 on SeqCIFAR-10, and 0.49-0.65 on DVSCIFAR-10. The misallocation is therefore not tied to a particular timestep or surrogate choice. These results lead to the following conclusion:

> **Surrogate gradients misallocate adversarial sensitivity.** Existing dynamics-agnostic surrogate functions cannot track consequential spike flip dynamics, misallocating backward sensitivity with forward adversarial response. This forms a key bottleneck in surrogate-based learning for SNNs.

4 METHOD

Motivated by the observed misallocation, we propose Flip-Aware Reallocation (FARE) for SNN robustness. FARE operates on the surrogate gradient, endowing it with awareness of perturbation-reachable and decision-relevant spike flips, and uses this information to reallocate backward sensitivity across membrane states. To ground this design, we first identify what conventional fixed surrogate functions fail to capture and explain how this failure gives rise to the observed misallocation.

Origin of misallocation. Ideally, if a surrogate gradient is to reflect the finite-step sensitivity of the hard-spike dynamics, its local response should best approximate the spike changes induced by finite membrane displacements, while placing greater emphasis on changes that are more relevant to the final decision. We make this requirement precise through a conditional least-squares criterion:

$$\psi^\star(z) = \arg\min_{\psi(z)\in\mathbb{R}} \mathbb{E}\Big[w_k \big(\Delta s_k - \psi(z) q_k\big)^2 \,\Big|\, z_k = z \Big] = \frac{\mathbb{E}[w_k q_k \Delta s_k \mid z_k = z]}{\mathbb{E}[w_k {q_k}^2 \mid z_k = z]}, \tag{11}$$

where, as defined in Eq. (8), z_k is the clean membrane-to-threshold distance, q_k is the membrane displacement, and Δs_k is the resulting spike flip. $w_k = |\partial M(x,y)/\partial s_k|$ is the absolute clean-point margin sensitivity to spike s_k. It measures the local importance of that spike to the decision margin. Eq. (11) gives the conditionally optimal surrogate response under this criterion. The numerator measures the decision-weighted alignment between membrane displacement and realized spike flips, states without a flip contribute zero because $\Delta s_k = 0$. The denominator normalizes this signal by the total decision-weighted displacement energy at the same threshold distance. Consequently, $\psi^\star(z)$ assigns stronger sensitivity where finite perturbations frequently induce decision-relevant flips relative to the observed displacement, and weaker sensitivity where membrane movement rarely changes the spike state. This criterion shows that an appropriate finite-step response depends on the conditional dynamics $p(q, \Delta s, w \mid z)$, including perturbation-induced membrane displacement, realized spike flips, and their relevance to the decision margin. Conventional hand-crafted surrogates, such as Triangle, Sigmoid, and Rectangle, prescribe a fixed coefficient profile $\psi(z)$ across membrane states. Although the local response $\psi(z)q_k$ contains the observed displacement, its allocation over z_k remains independent of whether that displacement induces a hard spike flip Δs_k and how strongly the flip affects the decision margin through w_k. The missing reallocation information must therefore identify where perturbations induce spike flips and weight these locations by their decision relevance.

From decision-relevant flips to response ranges. Eq. (11) shows that the conditionally optimal response depends on where membrane displacements induce decision-relevant spike flips. Therefore, the threshold distances of these flips identify the membrane regions that should receive surrogate sensitivity. Under the unit-mass convention applied in Eq. (5), the response range determines how far this fixed sensitivity mass extends: a narrow range poorly covers reachable flips, whereas an overly broad range disperses sensitivity over less relevant states. Therefore, FARE first converts weighted flip-distance statistics into adaptive response ranges on the two sides of the threshold. We partition the observed spike flips induced by the membrane displacement q_k into two directions:

$$\mathcal{F}_- = \{k : z_k < 0,\ \Delta s_k = +1\}, \qquad \mathcal{F}_+ = \{k : z_k \geq 0,\ \Delta s_k = -1\}. \tag{12}$$

Here, $\mathcal{F}_-$ and $\mathcal{F}_+$ contain silent-to-spike and spike-to-silent flips, respectively. Since these flips are induced by the observed displacement q_k, their clean distances $|z_k|$ reveal how far the surrogate response should extend on each side of the threshold. Building on flip statistics, FARE updates the adaptive response ranges through weighted quantiles and an exponential moving average, as follows:

$$\widehat{\beta}_\pm = Q_\xi^w\big(\{(|z_k|, w_k) : k \in \mathcal{F}_\pm\}\big), \qquad \beta_\pm \leftarrow \rho\beta_\pm + (1-\rho)\,\mathrm{clip}\Big(\widehat{\beta}_\pm, \beta_{\min}, \beta_{\max}\Big). \tag{13}$$

Here, Q_ξ^w denotes the weighted quantile (Hyndman & Fan, 1996). By Eq. (13), FARE uses $\beta_\pm(\Delta s, w)$ to dynamically track the response ranges required by the observed spike flips on the two sides of the firing threshold. Additional details for response ranges $\beta_\pm(\Delta s, w)$ are provided in Appendix C.

From response ranges to fixed-mass reallocation. Unlike the adversarially agnostic fixed range β_0 prescribed by conventional surrogates such as Triangle, the adaptive ranges $\beta_\pm(\Delta s, w)$ collect information from the model's adversarial dynamics. To incorporate this dynamic information into backward propagation, FARE applies $\beta_\pm(\Delta s, w)$ to the original surrogate function ψ_0, producing a flip-aware response branch $\psi_0(z, \beta_\pm(\Delta s, w))$. On the other hand, due to original response $\psi_0(z, \beta_0)$ preserves the established threshold-local credit-assignment signal that enables task learning, the resulting surrogate gradient under FARE combines the original and flip-aware responses by

$$\widetilde{\psi}(z) = [\psi_0(z, \beta_0) + \psi_0(z, \beta_\pm(\Delta s, w))]/2. \tag{14}$$

Since the original and flip-aware profiles both have unit mass, their sum has a total mass of two, and the factor $1/2$ restores the unit-mass convention. Consequently, FARE preserves the original threshold-local signal while using empirical samples from the dynamics $p(q, \Delta s, w \mid z)$ underlying Eq. (11) to reallocate the sensitivity toward perturbation-reachable, decision-relevant flip regions.

FARE in a nutshell. FARE converts perturbation-reachable, decision-relevant spike flips into adaptive response ranges on both sides of the firing threshold. By combining the resulting flip-aware response with the original surrogate, it reallocates backward sensitivity toward important flip regions while preserving the total sensitivity mass and the hard-spike forward dynamics.

5 EXPERIMENT

In this section, we evaluate FARE on five datasets spanning static, sequential, event-based, and large-scale vision. We compare its robustness with SOTA methods, followed by analyses of attack access, sensitivity reallocation, adversarial trajectories, and ablation study for key design choices.

5.1 MAIN RESULTS AND MECHANISM ANALYSIS

Table 1: White-box comparison with SOTA methods (%). The best in each column is highlighted. All methods use VGG11 architecture except SR (Liu et al., 2024), which uses WRN16. AT (Kundu et al., 2021) is self-implemented. *N/A* indicates collapsed models without analytical significance.

Method / Surrogate	FARE	CIFAR-10			CIFAR-100			DVSCIFAR-10		
		Clean	FGSM	PGD	Clean	FGSM	PGD	Clean	FGSM	PGD
TGO (Wang et al., 2026b)	-	88.79	51.40	11.67	69.47	17.20	2.30	-	-	-
FEEL (Xu et al., 2024)	-	93.29	44.96	28.35	**73.79**	9.60	2.04	-	-	-
STOD (Zhang et al., 2026a)	-	91.43	55.80	32.97	71.00	26.26	13.13	74.70	66.10	60.40
DLIF (Ding et al., 2024a)	-	92.39	15.24	0.23	70.51	8.72	0.77	-	-	-
DSD (Zhang et al., 2026b)	-	90.21	55.86	30.21	70.26	23.81	8.09	75.10	65.80	61.20
SR (Liu et al., 2024)	-	85.91	34.79	30.54	60.37	25.76	**19.76**	75.60	64.60	61.20
AT (Kundu et al., 2021)	-	91.96	38.20	14.07	69.69	16.31	8.49	74.30	42.50	37.90
Rectangle	✗	87.71	7.55	0.01	68.62	3.76	0.07	*N/A*	*N/A*	*N/A*
	✓	85.91	42.17	24.75	65.09	20.75	9.25	*N/A*	*N/A*	*N/A*
Sigmoid	✗	90.29	7.61	0.01	70.16	3.73	0.16	71.90	13.40	2.10
	✓	88.48	49.79	29.91	68.90	24.63	12.82	70.20	58.70	50.00
Triangle	✗	**93.75**	8.19	0.03	72.39	4.55	0.19	**76.30**	17.20	5.00
	✓	93.54	**58.70**	**33.89**	71.99	**29.71**	14.07	**76.30**	**67.80**	**62.90**

White-box comparison with SOTA methods. Table 1 compares FARE with SOTA SNN robustness methods on CIFAR-10, CIFAR-100, and DVSCIFAR-10. With Triangle as the base surrogate, FARE increases FGSM accuracy over the SNN baseline by 25.16-50.60% and PGD accuracy by 13.88-57.90%, while changing clean accuracy by at most 0.40%, this near-baseline clean accuracy is unmatched by other methods. On CIFAR-10, FARE achieves 58.70% under FGSM and 33.89% under PGD, exceeding the strongest competing results by 2.84% and 0.92%, respectively. On CIFAR-100, it obtains the highest FGSM accuracy of 29.71% and reaches 14.07% under PGD, outperforming all listed VGG11-based competitors. On DVSCIFAR-10, FARE preserves the 76.30% clean accuracy of the baseline while achieving the best FGSM and PGD accuracies of 67.80% and 62.90%. Consistent gains also appear across every valid Rectangle and Sigmoid configuration, where FARE substantially improves both FGSM and PGD accuracy. These results establish that FARE delivers strong white-box robustness while remaining consistently compatible with different mainstream surrogate functions.

Sequential and large-scale tasks. We evaluate FARE on SeqCIFAR-10 and ImageNet, which represent long-sequence processing and large-scale tasks, respectively. As shown in Table 2, using Triangle, FARE improves SeqCIFAR-10 accuracy by 22.66% under FGSM and 12.99% under PGD, with only 1.26% change in clean accuracy. On ImageNet, it improves FGSM and PGD accuracy by 16.59% and 7.68%, respectively, while retaining 56.14% clean accuracy. Every valid Rectangle and Sigmoid configuration exhibits the same robust improvement. These results, together with Table 1, demonstrate that FARE remains effective across substantially different temporal scales, dataset sizes, and mainstream surrogate functions.

Table 2: Performance in sequential and large-scale tasks. *N/A* indicates collapsed models without analytical significance.

Surrogate	FARE	SeqCIFAR-10			ImageNet		
		Clean	FGSM	PGD	Clean	FGSM	PGD
Rectangle	✗	53.02	1.48	0.00	*N/A*	*N/A*	*N/A*
	✓	51.70	19.69	7.36	*N/A*	*N/A*	*N/A*
Sigmoid	✗	59.06	2.71	0.01	52.38	2.87	0.00
	✓	56.08	21.83	9.09	49.00	16.61	4.50
Triangle	✗	69.46	4.85	0.04	57.84	4.99	0.01
	✓	68.20	27.51	13.03	56.14	21.58	7.69

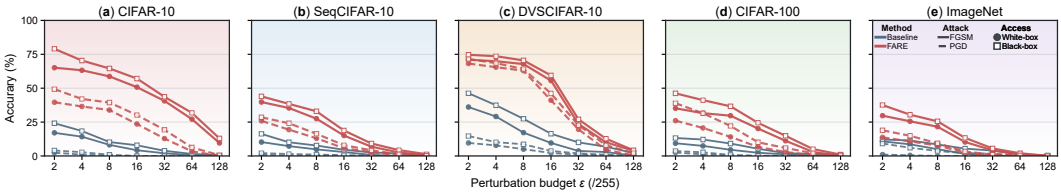

Figure 4: White- and black-box attacks comparison. Detailed results provided in Appendix F.

Black-box robustness comparisons. Figure 4 compares FARE with the SNN baseline under FGSM and PGD attacks across five datasets. We vary the perturbation budget to examine how their performance changes with attack strength under both white-box and black-box access. Across all five datasets, both attacks, and both access settings, FARE consistently outperforms the baseline at $\varepsilon \in \{2, 4, 8, 16\}/255$, showing that its gains extend beyond a single evaluation budget. The advantage is particularly pronounced under PGD, where baseline accuracy rapidly approaches zero on several datasets while FARE retains substantially higher accuracy. At $\varepsilon = 8/255$, for example, FARE achieves white-box/black-box PGD accuracies of 33.89%/39.40% on CIFAR-10 and 62.90%/64.10% on DVSCIFAR-10, compared with 0.03%/0.89% and 5.00%/8.60% for the baseline. The gains also extend to large-scale recognition, with ImageNet PGD accuracy increasing from 0.01%/3.67% to 7.69%/9.33%. Overall, the results demonstrate broad robustness improvements across attack strengths and access settings, rather than an advantage confined to one attack configuration.

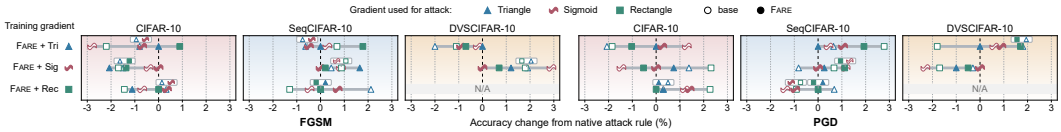

Figure 5: Robustness under alternative backward rules. Detailed results provided in Appendix F.

White-box robustness under alternative surrogate-gradient access. Since FARE directly modifies the surrogate gradients used in backpropagation, we examine whether its white-box robustness persists when attackers use alternative backward rules to generate adversarial perturbations. With full access to each fixed model, the attacker generates FGSM and PGD perturbations using Triangle, Sigmoid, Rectangle, and their FARE-adjusted counterparts. Model parameters and hard-spike forward computation remain unchanged. Figure 5 reports accuracy changes relative to the model's native FARE rule, with negative values indicating stronger attacks. Across all valid configurations, switching attack backward rules causes no abrupt accuracy collapse, with maximum reductions of 2.80% and 2.30% under FGSM and PGD, respectively. For example, the PGD accuracy of the FARE-Triangle model decreases from 33.89% to a minimum of 31.82% on CIFAR-10. Additionally, the strongest rule varies across configurations. These results support that FARE's robustness reflects resilience of the learned hard-spike model rather than reliance on its native surrogate rule during attack generation.

Sensitivity reallocation toward spike-flip regions. We examine whether FARE improves the sensitivity allocation identified in our motivation study. Figure 6 compares the normalized surrogate-response distributions before and after applying FARE and quantifies their alignment with actual spike flips using d_{TV} from Eq. (10). Across the displayed configurations, FARE reshapes the surrogate response to more closely follow the concentration and spread of the flip distribution, increasing their

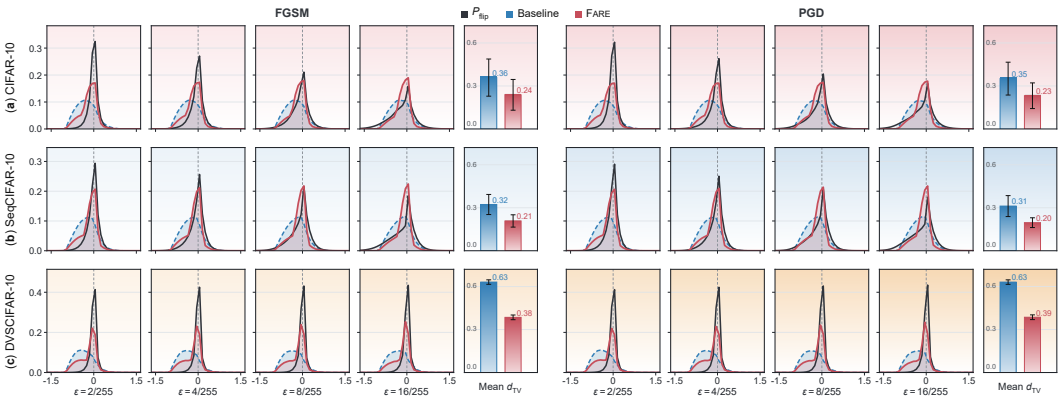

Figure 6: Sensitivity allocation before and after FARE. For clarity, only baseline P_{flip} is shown since the baseline and FARE visually produce completely overlapping flip profiles. The bar panels report the mean d_{TV} across four perturbation budgets, with error bars indicating the standard deviation.

overlap across membrane states. Quantitatively, FARE reduces d_{TV} in all configurations, with the overall mean decreasing from 0.434 to 0.274 (average across all configurations), a 36.9% reduction. The improvement holds on average for every dataset under both FGSM and PGD. The consistency across attacks and budgets confirms that the observed robustness gains are accompanied by the intended mechanism: a realignment of backward sensitivity with hard-forward spike dynamics.

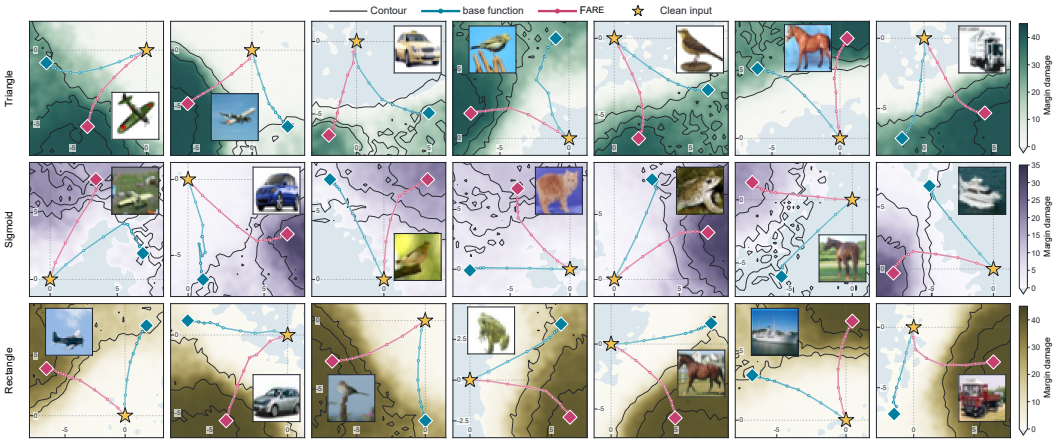

Figure 7: PGD trajectories. Curves connect projected iterates, stars mark the clean input at (0,0).

PGD trajectory analysis. Figure 7 examines how sensitivity reallocation changes adversarial search on specific examples across surrogate functions. In these examples, the base function directs PGD toward relatively low-damage regions, whereas FARE reaches regions with substantially greater realized margin damage. These trajectories show that reallocation can translate into more effective discovery of vulnerable directions in the actual hard-spike network. Together with the allocation analysis, they support the design rationale of FARE: decision-relevant spike-flip feedback provides a more informative backward sensitivity signal for identifying adversarial vulnerabilities in attacks.

5.2 Ablation Studies

Choice of weighted quantile. The quantile ξ in Eq. (13) determines the fraction of decision-weighted flip mass covered by the adaptive response ranges. As shown in Figure 8, increasing ξ from 0.25 to 0.80 substantially improves adversarial accuracy across all three datasets, highlighting the importance of sufficiently covering consequential flip regions. The default $\xi = 0.80$ achieves the highest PGD accuracy among the tested quantiles while preserving clean accuracy close to the baseline. Broader coverage also offers a surprisingly favorable trade-off. For example, on DVSCIFAR-10, $\xi = 0.95$ increases clean accuracy from 76.30% to 77.00%, alongside improvements from 17.20% to 68.30% under FGSM and from 5.00% to 62.00% under PGD. Thus, appropriate sensitivity reallocation can improve adversarial performance without sacrificing, and even while enhancing, clean performance.

Role of the original surrogate response. We remove the original response component $\psi_0(z;\beta_0)$ from the fusion in Eq. (14) and retain only the dynamic response $\psi_0(z,\beta_\pm(\Delta s,w))$ (dynamic SG only). Figure 8 shows that this configuration yields lower clean and adversarial accuracy than the complete FARE across all three datasets. PGD accuracy drops from 33.89% to 5.06% on CIFAR-10, while clean and PGD accuracy on DVSCIFAR-10 fall from 76.30% and 62.90% to 38.00% and 0.10%, a complete collapse, respectively. These results support the complementary roles of the two components: the original response preserves threshold-local learning signals, while the dynamic response adjusts sensitivity coverage using actual spike-flip feedback. Their combination supports effective learning on clean samples while reallocating sensitivity toward perturbation-relevant states to improve adversarial robustness.

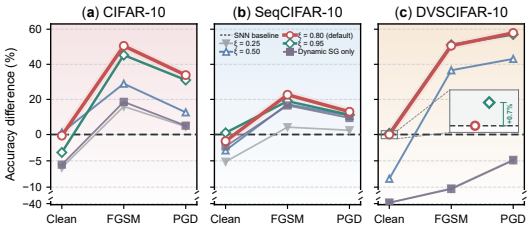

Figure 8: Ablation comparisons. Colored curves report accuracy differences from the SNN baseline.

6 Related Work

Adversarial robustness in spiking neural networks. Adversarial training, robustness-accuracy trade-offs, and local linearization provide complementary foundations for robust learning (Madry et al., 2017; Zhang et al., 2019; Qin et al., 2019). SNN defenses build on crafted perturbations and regularized adversarial training (Kundu et al., 2021; Ding et al., 2022), gradient sparsity (Liu et al., 2024), and neuronal stability and encoding mechanisms (Ding et al., 2024a; Xu et al., 2024). Recent methods further address threshold-related spike changes, temporal gradient collinearity, and dominant gradient directions (Wang et al., 2026b; Zhang et al., 2026a;b). Meanwhile, gradient-obfuscation studies and alternative-gradient attacks highlight the importance of backward rules for reliable evaluation (Athalye et al., 2018; Lun et al., 2025; Wang et al., 2026a). Different from these approaches, our work revisits SNN vulnerability from a new perspective of surrogate gradients and introduces FARE, advancing surrogate-function design as an avenue for improving robustness.

Surrogate-function-based design. Binary networks and straight-through estimator studies establish the importance of backward approximations for discrete optimization (Hubara et al., 2016; Yin et al., 2019). SNN research further examines surrogate learning, shape, scale, and theoretical foundations (Neftci et al., 2019; Zenke & Vogels, 2021; Gygax & Zenke, 2025). Adaptive approaches include shape and training-stage adaptation (Li et al., 2021; Guo et al., 2022), membrane-distribution-aware widths (Lian et al., 2023), learnable relaxation (Wang et al., 2023), and surrogate modules (Deng et al., 2023). Building on surrogate adaptation, FARE grounds response ranges in observed adversarial flips weighted by decision relevance. It combines adapted and original responses to redistribute sensitivity while preserving total response mass, enabling seamless integration with existing surrogate functions while preserving their role in the original training framework.

7 Discussion

Conclusion. This work identifies adversarial sensitivity misallocation as a limitation of conventional surrogate gradients in SNNs. Through model-level damage comparisons, optimal scale control, and membrane-state distribution analysis, we show that surrogate responses remain misaligned with perturbation-induced spike flips even after removing overall response scale. To address this, we propose FARE, which derives adaptive response ranges from decision-weighted spike flips and combines the resulting response with the original surrogate profile. This design reallocates backward sensitivity while preserving total response mass and hard-spike forward computation. Experiments demonstrate improved robustness in multiple attack access with limited clean-accuracy trade-offs. Mechanism analyses and ablations further validate the effectiveness of FARE component designs.

Broader implications and limitations. FARE integrates with existing surrogate functions without additional inference overhead. Although robustness improvements are accompanied by some loss of clean accuracy in most settings, FARE improves both clean and adversarial accuracy in selected configurations. This suggests that better sensitivity allocation can alleviate their trade-off, motivating future work to extend these joint benefits across tasks. The main limitation of FARE is the additional training time and peak memory. A detailed evaluation for training costs is provided in Appendix H.

Reproducibility Statement

The complete anonymized code is provided in the supplementary material. All datasets used in this work, including CIFAR-10, CIFAR-100, DVSCIFAR-10, SeqCIFAR-10, and ImageNet, are publicly accessible. Dataset descriptions and preprocessing protocols are detailed in the appendix, together with the experimental hardware and all hyperparameter settings. We provide these resources to enable future researchers to reproduce our results under the same or similar experimental conditions.

References

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*, pp. 274–283. PMLR, 2018.

Andy Brock, Soham De, Samuel L Smith, and Karen Simonyan. High-performance large-scale image recognition without normalization. In *International conference on machine learning*, pp. 1059–1071. PMLR, 2021.

Tong Bu, Jianhao Ding, Zecheng Hao, and Zhaofei Yu. Rate gradient approximation attack threats deep spiking neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7896–7906, 2023.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pp. 39–57. Ieee, 2017.

Benjamin Cramer, Sebastian Billaudelle, Simeon Kanya, Aron Leibfried, Andreas Grübl, Vitali Karasenko, Christian Pehle, Korbinian Schreiber, Yannik Stradmann, Johannes Weis, et al. Surrogate gradients for analog neuromorphic computing. *Proceedings of the National Academy of Sciences*, 119(4):e2109194119, 2022.

Mike Davies, Narayan Srinivasa, Tsung-Han Lin, Gautham Chinya, Yongqiang Cao, Sri Harsha Choday, Georgios Dimou, Prasad Joshi, Nabil Imam, Shweta Jain, et al. Loihi: A neuromorphic manycore processor with on-chip learning. *Ieee micro*, 38(1):82–99, 2018.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Shikuang Deng, Hao Lin, Yuhang Li, and Shi Gu. Surrogate module learning: Reduce the gradient error accumulation in training spiking neural networks. In *International Conference on Machine Learning*, pp. 7645–7657. PMLR, 2023.

Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

Jianhao Ding, Tong Bu, Zhaofei Yu, Tiejun Huang, and Jian Liu. Snn-rat: Robustness-enhanced spiking neural network through regularized adversarial training. *Advances in Neural Information Processing Systems*, 35:24780–24793, 2022.

Jianhao Ding, Zhiyu Pan, Yujia Liu, Zhaofei Yu, and Tiejun Huang. Robust stable spiking neural networks. *arXiv preprint arXiv:2405.20694*, 2024a.

Jianhao Ding, Zhaofei Yu, Tiejun Huang, and Jian K Liu. Enhancing the robustness of spiking neural networks with stochastic gating mechanisms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 492–502, 2024b.

Steve K Esser, Rathinakumar Appuswamy, Paul Merolla, John V Arthur, and Dharmendra S Modha. Backpropagation for energy-efficient neuromorphic computing. *Advances in neural information processing systems*, 28, 2015.

Wei Fang, Zhaofei Yu, Yanqi Chen, Tiejun Huang, Timothée Masquelier, and Yonghong Tian. Deep residual learning in spiking neural networks. *Advances in Neural Information Processing Systems*, 34:21056–21069, 2021a.

Wei Fang, Zhaofei Yu, Yanqi Chen, Timothée Masquelier, Tiejun Huang, and Yonghong Tian. Incorporating learnable membrane time constant to enhance learning of spiking neural networks. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2641–2651. IEEE, 2021b.

Wei Fang, Zhaofei Yu, Zhaokun Zhou, Ding Chen, Yanqi Chen, Zhengyu Ma, Timothée Masquelier, and Yonghong Tian. Parallel spiking neurons with high efficiency and ability to learn long-term dependencies. *Advances in Neural Information Processing Systems*, 36:53674–53687, 2023.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Yufei Guo, Yuanpei Chen, Liwen Zhang, Xiaode Liu, Yinglei Wang, Xuhui Huang, and Zhe Ma. Im-loss: information maximization loss for spiking neural networks. *Advances in neural information processing systems*, 35:156–166, 2022.

Julia Gygax and Friedemann Zenke. Elucidating the theoretical underpinnings of surrogate gradient learning in spiking neural networks. *Neural Computation*, 37(5):886–925, 2025.

Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized neural networks. *Advances in neural information processing systems*, 29, 2016.

Rob J Hyndman and Yanan Fan. Sample quantiles in statistical packages. *The American Statistician*, 50(4):361–365, 1996.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Souvik Kundu, Massoud Pedram, and Peter A Beerel. Hire-snn: Harnessing the inherent robustness of energy-efficient deep spiking neural networks by training with crafted input noise. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5209–5218, 2021.

Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016.

Hongmin Li, Hanchao Liu, Xiangyang Ji, Guoqi Li, and Luping Shi. Cifar10-dvs: an event-stream dataset for object classification. *Frontiers in neuroscience*, 11:244131, 2017.

Yuhang Li, Yufei Guo, Shanghang Zhang, Shikuang Deng, Yongqing Hai, and Shi Gu. Differentiable spike: Rethinking gradient-descent for training spiking neural networks. *Advances in neural information processing systems*, 34:23426–23439, 2021.

Shuang Lian, Jiangrong Shen, Qianhui Liu, Ziming Wang, Rui Yan, and Huajin Tang. Learnable surrogate gradient for direct training spiking neural networks. In *Ijcai*, pp. 3002–3010, 2023.

Yujia Liu, Tong Bu, Jianhao Ding, Zecheng Hao, Tiejun Huang, and Zhaofei Yu. Enhancing adversarial robustness in snns with sparse gradients. *arXiv preprint arXiv:2405.20355*, 2024.

Li Lun, Kunyu Feng, Qinglong Ni, Ling Liang, Yuan Wang, Ying Li, Dunshan Yu, and Xiaoxin Cui. Towards effective and sparse adversarial attack on spiking neural networks via breaking invisible surrogate gradients. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3540–3551. IEEE, 2025.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Emre O Neftci, Hesham Mostafa, and Friedemann Zenke. Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks. *IEEE Signal Processing Magazine*, 36(6):51–63, 2019.

Chongli Qin, James Martens, Sven Gowal, Dilip Krishnan, Krishnamurthy Dvijotham, Alhussein Fawzi, Soham De, Robert Stanforth, and Pushmeet Kohli. Adversarial robustness through local linearization. *Advances in neural information processing systems*, 32, 2019.

Andrew Ross and Finale Doshi-Velez. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Kaushik Roy, Akhilesh Jaiswal, and Priyadarshini Panda. Towards spike-based machine intelligence with neuromorphic computing. *Nature*, 575(7784):607–617, 2019.

Jihang Wang, Dongcheng Zhao, Ruolin Chen, Qian Zhang, and Yi Zeng. Towards reliable evaluation of adversarial robustness for spiking neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20691–20700, 2026a.

Shuai Wang, Malu Zhang, Yulin Jiang, Dehao Zhang, Ammar Belatreche, Yu Liang, Yimeng Shan, Zijian Zhou, Yang Yang, and Haizhou Li. Robust spiking neural networks against adversarial attacks. *arXiv preprint arXiv:2602.20548*, 2026b.

Ziming Wang, Runhao Jiang, Shuang Lian, Rui Yan, and Huajin Tang. Adaptive smoothing gradient learning for spiking neural networks. In *International conference on machine learning*, pp. 35798–35816. PMLR, 2023.

Yujie Wu, Lei Deng, Guoqi Li, Jun Zhu, and Luping Shi. Spatio-temporal backpropagation for training high-performance spiking neural networks. *Frontiers in neuroscience*, 12:331, 2018.

Mengting Xu, De Ma, Huajin Tang, Qian Zheng, and Gang Pan. Feel-snn: Robust spiking neural networks with frequency encoding and evolutionary leak factor. *Advances in Neural Information Processing Systems*, 37:91930–91950, 2024.

Bojian Yin, Federico Corradi, and Sander M Bohté. Accurate and efficient time-domain classification with adaptive spiking recurrent neural networks. *Nature Machine Intelligence*, 3(10):905–913, 2021.

Penghang Yin, Jiancheng Lyu, Shuai Zhang, Stanley Osher, Yingyong Qi, and Jack Xin. Understanding straight-through estimator in training activation quantized neural nets. *arXiv preprint arXiv:1903.05662*, 2019.

Friedemann Zenke and Tim P Vogels. The remarkable robustness of surrogate gradient learning for instilling complex function in spiking neural networks. *Neural computation*, 33(4):899–925, 2021.

Desong Zhang, Jia Hu, and Geyong Min. Breaking gradient temporal collinearity for robust spiking neural networks. In *The Fourteenth International Conference on Learning Representations*, 2026a.

Desong Zhang, Jia Hu, and Geyong Min. Robustify spiking neural networks via dominant singular deflation under heterogeneous training vulnerability. In *The Fourteenth International Conference on Learning Representations*, 2026b.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pp. 7472–7482. PMLR, 2019.

A SURROGATE FUNCTIONS

We consider three commonly used surrogate functions with distinct sensitivity profiles: Triangle, Sigmoid, and Rectangle (Neftci et al., 2019; Wu et al., 2018). Let $z = u - u_{\text{th}}$ denote the membrane-to-threshold distance and $\beta > 0$ control the response width. For each surrogate, we denote its smooth primitive by $G(z, \beta)$ and its derivative used during backward propagation by $\psi(z, \beta) = \partial G(z, \beta)/\partial z$. The actual forward pass always follows the hard-spike rule $H(z)$. All three derivatives satisfy the unit-mass convention $\int_{-\infty}^{+\infty} \psi(z, \beta)\, \mathrm{d}z = 1$.

Triangle. The Triangle surrogate is the derivative of a piecewise-quadratic approximation to the Heaviside function:

$$G_{\text{tri}}(z, \beta) = \begin{cases} 0, & z \le -\beta, \\ \dfrac{(z+\beta)^2}{2\beta^2}, & -\beta < z < 0, \\ 1 - \dfrac{(\beta - z)^2}{2\beta^2}, & 0 \le z < \beta, \\ 1, & z \ge \beta, \end{cases} \qquad \psi_{\text{tri}}(z, \beta) = \frac{1}{\beta} \max\left(0, 1 - \frac{|z|}{\beta}\right). \tag{15}$$

Sigmoid. The Sigmoid surrogate uses the logistic function as its smooth primitive:

$$G_{\text{sig}}(z, \beta) = \sigma\left(\frac{z}{\beta}\right) = \frac{1}{1 + \exp(-z/\beta)}, \qquad \psi_{\text{sig}}(z, \beta) = \frac{1}{\beta} \sigma\left(\frac{z}{\beta}\right)\left[1 - \sigma\left(\frac{z}{\beta}\right)\right]. \tag{16}$$

Rectangle. The Rectangle surrogate is obtained from a piecewise-linear approximation to the Heaviside function:

$$G_{\text{rec}}(z, \beta) = \begin{cases} 0, & z \le -\beta, \\ \dfrac{z+\beta}{2\beta}, & -\beta < z < \beta, \\ 1, & z \ge \beta, \end{cases} \qquad \psi_{\text{rec}}(z, \beta) = \begin{cases} \dfrac{1}{2\beta}, & |z| \le \beta, \\ 0, & |z| > \beta. \end{cases} \tag{17}$$

B ADVERSARIAL ATTACKS

We consider untargeted attacks that maximize the classification loss $\mathcal{L}$. All reported experiments use the ℓ_∞ threat model unless otherwise specified. Let $\mathcal{X}$ denote the valid input domain and $\Pi_{\mathcal{S}}(\cdot)$ the projection onto a feasible set $\mathcal{S}$.

Fast Gradient Sign Method. The Fast Gradient Sign Method (FGSM) (Goodfellow et al., 2014) performs a single gradient-ascent step with step size equal to the perturbation budget:

$$x_{\text{adv}} = \Pi_{\mathcal{X}}\left[x + \varepsilon\ \text{sign}\left(\widetilde{\nabla}_x^{\psi} \mathcal{L}\big(f(x), y\big)\right)\right], \qquad \delta_{\text{FGSM}} = x_{\text{adv}} - x, \tag{18}$$

where $\widetilde{\nabla}_x^{\psi}$ denotes the input gradient computed through surrogate function ψ. The projection ensures that the resulting adversarial example remains in the valid input domain.

Projected Gradient Descent. Projected Gradient Descent (PGD) (Madry et al., 2017) extends FGSM by applying multiple gradient-ascent steps within the perturbation set. Starting from $x_{\text{adv}}^{(0)}$, its r-th update is

$$x_{\text{adv}}^{(r+1)} = \Pi_{\mathcal{X} \cap \mathcal{B}_\infty(x, \varepsilon)}\left[x_{\text{adv}}^{(r)} + \alpha\ \text{sign}\left(\widetilde{\nabla}^{\psi}_{x_{\text{adv}}^{(r)}} \mathcal{L}\big(f(x_{\text{adv}}^{(r)}), y\big)\right)\right], \tag{19}$$

where α is the attack step size and $\mathcal{B}_\infty(x, \varepsilon) = \{x' \colon \|x' - x\|_\infty \le \varepsilon\}$. Standard robustness evaluation uses a random initialization within this perturbation set and repeats Eq. (19) for the prescribed number of iterations.

White-box and black-box evaluation. Under white-box evaluation, adversarial examples are generated using gradients from the evaluated model itself. Under black-box evaluation, they are generated by a separately trained source model and then transferred to the target model without accessing its gradients.

Hyperparameters. For these attack methods, we set $\varepsilon = 8/255$ for all experimental cases unless otherwise specified. For PGD, step size $\alpha = 0.01$ and step number $K = 10$.

C Additional Details of FARE

Periodic flip collection. During training, FARE collects spike-flip statistics every K mini-batches. Given a clean mini-batch x, we generate a bounded calibration counterpart $x^{\mathrm{cal}} = x + \delta^{\mathrm{cal}}$ using FGSM with the current model. The clean trajectory provides the threshold distance z_k and decision relevance w_k, while comparison with the calibration trajectory identifies the spike flip Δs_k. Statistics are pooled across samples, spiking layers, and timesteps to estimate one global pair of response ranges, β_- and β_+.

Weighted range estimation. For each flip direction, the weighted quantile is implemented as

$$Q_\xi^w(\mathcal{F}_\pm) = \inf\left\{ r \geq 0 : \frac{\sum_{k\in\mathcal{F}_\pm} w_k \mathbb{I}(|z_k| \leq r)}{\sum_{k\in\mathcal{F}_\pm} w_k} \geq \xi \right\}. \tag{20}$$

We set $\xi = 0.8$, so the instantaneous estimate $\widehat{\beta}_\pm$ covers 80% of the observed decision-weighted flip mass on the corresponding side of the threshold. This estimator emphasizes flip distances associated with greater decision relevance while reducing the influence of low-importance tail observations.

Smoothing and numerical bounds. The running ranges are initialized as $\beta_- = \beta_+ = 1.0$, matching the initial width of the base surrogate. Each instantaneous estimate is clipped to $[\beta_{\min}, \beta_{\max}] = [0.05, 8.0]$ and then incorporated using an EMA coefficient of $\rho = 0.9$. Thus, each update retains 90% of the previous range and incorporates 10% of the current estimate. The clipping bounds are fixed numerical safeguards that prevent near-zero or excessively broad response ranges, they are chosen once and remain unchanged across datasets, architectures, and surrogate functions.

Calibration budgets. The calibration perturbation is used to expose perturbation-reachable spike flips rather than to optimize an adversarial training objective. We use $\varepsilon_{\mathrm{cal}} = 8/255$ for ImageNet and $2/255$ for other datasets.

D Datasets

CIFAR-10. The CIFAR-10 dataset (Krizhevsky et al., 2009) consists of 60,000 color images, each of size 32×32 pixels, divided into 10 different classes, such as airplanes, cars, birds, cats, and dogs. Each class has 6,000 images, with 50,000 images used for training and 10,000 for testing. Normalization, random horizontal flipping, random cropping with 4 padding, and CutOut (DeVries & Taylor, 2017) are applied for data augmentation.

CIFAR-100. The CIFAR-100 dataset (Krizhevsky et al., 2009) consists of 60,000 color images, each of size 32×32 pixels, categorized into 100 different classes. Each class contains 600 images, with 500 used for training and 100 for testing. The same processing methods as for dataset CIFAR-10 are applied to dataset CIFAR-100.

SeqCIFAR-10. We follow the column-wise sequential CIFAR-10 protocol used by Fang et al. (Fang et al., 2023) for evaluating long-term dependencies in SNNs. It retains the original CIFAR-10 classes and splits but converts each $3 \times 32 \times 32$ image into a left-to-right sequence of vertical stripes. The original $T = 32$ setting presents one $3 \times 32 \times 1$ column per time step. We additionally construct $T = 16$ and $T = 8$ variants by grouping every two and four adjacent columns, respectively, yielding samples of shape $T \times 3 \times 32 \times (32/T)$. This lossless partition preserves every pixel without resizing or duplication. The standard CIFAR-10 preprocessing is applied to the complete image before partitioning. Varying T therefore changes the temporal horizon and per-step spatial granularity while preserving the underlying image content.

DVSCIFAR-10. The DVSCIFAR-10 dataset (Li et al., 2017) is a neuromorphic version of the traditional CIFAR-10 dataset. DVSCIFAR-10 captures the visual information using a Dynamic Vision Sensor (DVS), which records changes in the scene as a series of asynchronous events rather than as a sequence of frames. The dataset consists of recordings of 10 object classes, corresponding to the original CIFAR-10 categories, with each object presented in front of a DVS camera under various conditions. The dataset contains 10,000 128×128 images, of which 9,000 are used as the training set and the remaining 1,000 as the test set.

ImageNet. We evaluate on the ILSVRC-2012 ImageNet dataset (Deng et al., 2009), which contains ∼1.28M training images and 50,000 validation images spanning 1,000 classes. Images are of variable

resolution, following common practice and our implementation, training augmentation includes RandomResizedCrop to 224×224, RandomHorizontalFlip, conversion to tensors, and channel-wise normalization. For test, images are resized to have a shorter side of 256 pixels and then center-cropped to 224×224 before applying the same normalization.

E Experimental Setup

E.1 Motivation Experiments

Residual measurement. We conduct the motivation analysis on clean-trained VGG11 checkpoints and freeze all model parameters during measurement. Hyperparameters are set the same as Table 3.

Global-scale control. For each configuration and perturbation budget, clean-correct samples are divided into disjoint calibration and held-out subsets according to their sample identifiers. The optimal positive scalar $c^\star$ in Eq. (6) is fitted on the calibration subset, while $\mathrm{Err}(c^\star)$ is evaluated on the held-out subset.

Scale-free allocation analysis. We consider Triangle and Sigmoid surrogates at $T \in \{8, 32\}$ and divide the threshold-distance interval $[-4, 4]$ into 80 equally spaced bins. The surrogate-response and spike-flip profiles are normalized separately before calculating d_{TV}.

E.2 Main Experiments

Table 3 lists the principal training settings used in the main experiments. CIFAR-10, SeqCIFAR-10, DVSCIFAR-10, and CIFAR-100 are trained on a single NVIDIA RTX 4070 Ti, while ImageNet is trained on a single NVIDIA H200 NVL. VGG11 is used for the four smaller datasets, and a spiking NF-ResNet18 (Brock et al., 2021) is used for ImageNet.

Table 3: Dataset-dependent configurations for the main experiments.

Dataset	Architecture	Timesteps	Epochs	Batch size	Weight decay	Hardware
CIFAR-10	VGG11	4	200	128	5×10^{-4}	RTX 4070 Ti
SeqCIFAR-10	VGG11	32	200	128	5×10^{-4}	RTX 4070 Ti
DVSCIFAR-10	VGG11	10	200	16	5×10^{-4}	RTX 4070 Ti
CIFAR-100	VGG11	4	200	128	5×10^{-4}	RTX 4070 Ti
ImageNet	Spiking NF-ResNet18	4	100	512	1×10^{-5}	H200 NVL

All models use LIF neurons with membrane time constant $\tau = 1.1$ and firing threshold $u_{\mathrm{th}} = 1.0$. We optimize the models using SGD with an initial learning rate of 0.1, momentum of 0.9, and cosine learning-rate annealing. The baseline and FARE use the same hard-forward architecture, temporal readout, data processing, and clean supervised objective within each dataset. Mixed-precision training is enabled, and the default random seed is 2026.

F Detailed Results

This section presents the detailed results of figures 4 (Table 4) and 5 (Table 5).

G AI Use Statement

AI tools assisted with English-language polishing and proofreading, figure design and color selection, literature searches, and drafting this statement.

H Training Cost Evaluation

Table 6 reports the training cost comparison between SNN baseline and FARE, where peak memory is measured using `torch.cuda.max_memory_allocated`. FARE increases peak allocated

Table 4: Detailed results (%) corresponding to Figure 4. WB and BB denote white-box and black-box access.

Dataset	Method	Attack	Access	ε (/255)							
				0	2	4	8	16	32	64	128
CIFAR-10	SNN baseline	FGSM	WB	93.75	17.06	14.22	8.19	4.21	1.99	0.58	0.00
	FARE	FGSM	WB	93.54	65.11	63.21	58.70	50.76	40.63	27.14	9.57
	SNN baseline	FGSM	BB	93.75	24.13	18.41	10.26	7.88	3.82	1.48	0.59
	FARE	FGSM	BB	93.54	79.02	70.38	64.55	57.10	43.65	31.76	12.94
	SNN baseline	PGD	WB	93.75	2.37	1.01	0.03	0.00	0.00	0.00	0.00
	FARE	PGD	WB	93.54	39.62	36.51	33.89	23.58	12.83	3.40	0.12
	SNN baseline	PGD	BB	93.75	4.01	2.80	0.89	0.02	0.00	0.00	0.00
	FARE	PGD	BB	93.54	49.30	42.04	39.40	30.19	19.27	6.26	0.58
SeqCIFAR-10	SNN baseline	FGSM	WB	69.46	10.25	7.25	4.85	2.88	0.75	0.02	0.00
	FARE	FGSM	WB	68.20	39.78	35.37	27.51	15.05	6.82	2.58	0.48
	SNN baseline	FGSM	BB	69.46	16.27	10.09	7.68	4.56	2.62	1.84	0.30
	FARE	FGSM	BB	68.20	43.96	38.42	32.95	18.78	9.18	4.30	1.31
	SNN baseline	PGD	WB	69.46	0.98	0.10	0.04	0.00	0.00	0.00	0.00
	FARE	PGD	WB	68.20	25.88	19.43	13.03	5.74	2.71	0.58	0.01
	SNN baseline	PGD	BB	69.46	2.10	1.43	1.01	0.63	0.12	0.00	0.00
	FARE	PGD	BB	68.20	28.52	24.12	16.35	7.96	4.12	1.27	0.27
DVSCIFAR-10	SNN baseline	FGSM	WB	76.30	36.10	29.10	17.20	9.60	3.70	2.90	0.20
	FARE	FGSM	WB	76.30	71.00	69.90	67.80	55.60	23.60	10.10	3.80
	SNN baseline	FGSM	BB	76.30	46.30	37.40	27.40	16.40	10.00	6.80	2.20
	FARE	FGSM	BB	76.30	74.66	73.60	70.50	59.40	27.00	12.70	4.20
	SNN baseline	PGD	WB	76.30	9.60	7.80	5.00	2.20	0.90	0.20	0.00
	FARE	PGD	WB	76.30	68.18	65.50	62.90	41.10	19.60	4.50	0.50
	SNN baseline	PGD	BB	76.30	14.70	10.20	8.60	3.70	1.90	0.90	0.10
	FARE	PGD	BB	76.30	71.51	68.50	64.10	46.10	22.70	7.90	2.10
CIFAR-100	SNN baseline	FGSM	WB	72.39	9.37	7.42	4.55	2.35	1.19	0.33	0.00
	FARE	FGSM	WB	71.99	35.17	31.55	29.71	20.22	11.21	2.96	0.04
	SNN baseline	FGSM	BB	72.39	13.26	12.11	9.16	5.31	2.07	1.47	0.49
	FARE	FGSM	BB	71.99	46.20	41.18	36.63	24.44	14.86	5.08	1.02
	SNN baseline	PGD	WB	72.39	2.53	1.15	0.19	0.02	0.00	0.00	0.00
	FARE	PGD	WB	71.99	26.03	20.65	14.07	6.66	2.93	0.47	0.01
	SNN baseline	PGD	BB	72.39	3.65	2.68	0.78	0.15	0.02	0.00	0.00
	FARE	PGD	BB	71.99	38.92	31.49	22.12	10.09	5.87	2.07	0.48
ImageNet	SNN baseline	FGSM	WB	57.84	10.75	8.59	4.99	1.56	0.14	0.01	0.00
	FARE	FGSM	WB	56.14	29.71	25.58	21.58	10.30	4.17	1.02	0.06
	SNN baseline	FGSM	BB	57.84	12.13	11.43	8.46	5.42	3.93	1.25	0.35
	FARE	FGSM	BB	56.14	37.58	30.44	25.55	13.34	5.55	1.98	0.12
	SNN baseline	PGD	WB	57.84	1.02	0.48	0.01	0.00	0.00	0.00	0.00
	FARE	PGD	WB	56.14	13.72	10.74	7.69	2.46	0.03	0.00	0.00
	SNN baseline	PGD	BB	57.84	9.19	6.05	3.67	2.02	1.79	0.05	0.00
	FARE	PGD	BB	56.14	18.91	14.89	9.33	3.29	0.99	0.47	0.02

memory by 93.9%-111.8% and average training time by 80.4%-235.9% across the five datasets. The additional cost comes from periodic calibration and decision-weighted spike-flip collection, with larger overhead on temporal inputs that require repeated processing of membrane trajectories. These operations are disabled after training, leaving the hard-spike inference graph, parameter count, and inference cost unchanged.

Table 5: Detailed results (%) corresponding to Figure 5. Shading marks each model's native FARE attack rule. *N/A* denotes a collapsed training configuration.

Dataset	**Attack**	**Training gradient**	**Attack backward rule**					
			Base			FARE		
			Triangle	Sigmoid	Rectangle	Triangle	Sigmoid	Rectangle
CIFAR-10	FGSM	FARE + Triangle	57.90	55.90	56.50	58.70	58.01	59.59
	FGSM	FARE + Sigmoid	48.30	49.37	48.10	47.72	49.79	48.42
	FGSM	FARE + Rectangle	42.49	41.48	40.73	41.06	42.52	42.17
	PGD	FARE + Triangle	31.82	35.18	32.04	33.89	34.15	32.86
	PGD	FARE + Sigmoid	31.29	28.45	32.22	30.65	29.91	29.38
	PGD	FARE + Rectangle	25.06	25.96	27.02	25.05	26.16	24.75
SeqCIFAR-10	FGSM	FARE + Triangle	26.88	27.79	28.20	27.51	27.02	29.29
	FGSM	FARE + Sigmoid	22.28	22.69	22.71	23.48	21.83	22.03
	FGSM	FARE + Rectangle	21.81	19.23	18.39	19.71	20.41	19.69
	PGD	FARE + Triangle	13.68	13.47	15.81	13.03	14.08	14.96
	PGD	FARE + Sigmoid	8.27	10.25	9.76	9.35	9.09	10.21
	PGD	FARE + Rectangle	8.04	5.98	6.45	7.37	6.40	7.36
DVSCIFAR-10	FGSM	FARE + Triangle	65.80	67.60	66.70	67.80	66.90	67.10
	FGSM	FARE + Sigmoid	60.60	61.60	60.50	59.90	58.70	59.40
	FGSM	FARE + Rectangle	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*
	PGD	FARE + Triangle	64.70	63.50	61.10	62.90	63.80	64.60
	PGD	FARE + Sigmoid	49.70	47.70	48.30	49.00	50.00	49.50
	PGD	FARE + Rectangle	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*

Table 6: Training-cost comparison between the SNN baseline and FARE. Experiments on ImageNet use an NVIDIA H200 NVL, while the remaining experiments use an NVIDIA RTX 4070 Ti.

Dataset	Method	Batch size	Batches/epoch	Peak memory (GB)	Time/epoch (s)
CIFAR-10	SNN baseline	128	390	0.9256	24.71
	FARE			1.9359	44.57
SeqCIFAR-10	SNN baseline	128	390	0.6742	54.72
	FARE			1.3072	144.32
DVSCIFAR-10	SNN baseline	16	562	3.9004	98.67
	FARE			8.2623	331.44
CIFAR-100	SNN baseline	128	390	0.9509	24.82
	FARE			1.9609	45.26
ImageNet	SNN baseline	512	2502	29.9057	525.56
	FARE			72.1185	928.19